REMARKS

Applicants are in receipt of the Office Action mailed March 25, 2004. Claims 1-90 remain pending in the application. Reconsideration is respectfully requested in light of the following remarks.

Section 102(e) Rejection:

The Office Action rejected claims 1, 9-11, 25, 27-28, 30-33, 34-39, 40, 41, 44, 46-49, 50, 51, 58-61, 62-66, 69, 70, 71, 75-77 and 84-90 were rejected under 35 U.S.C. § 102(e) as being anticipated by Meltzer et al. (U.S. Patent 6,542,912) (hereinafter "Meltzer").

Regarding claim 1, the Examiner contends that Meltzer discloses a method comprising, "a process executing within a virtual machine providing a first computer programming language object to a compilation process of the virtual machine, wherein the first object is an instance of a class in the computer programming language; the compilation process of the virtual machine converting the first object into a data representation language representation of the first object; wherein the data representation language representation of the first object is configured for use in generating a copy of the first object." Applicants disagree with the Examiner's interpretation of Meltzer.

Meltzer teaches a document-based transaction architecture for connecting businesses with customers, suppliers, and trading partners wherein companies exchanges information using self-defining documents, such as XML documents. Business interface definition (BID) documents describe how the XML documents are structured and how the information in the XML documents is to be used when conducting business transactions. When an XML document is received, it is parsed and a Java object is created that includes information from the XML document. This Java object is then forwarded to various enterprise or commercial functions. Any output is used to create an output XML document for return. (Meltzer, column 2, lines 45-67, column 5, lines 11-

32). The XML document in Meltzer represents information, not an object. Although Meltzer does create an object from the XML document, the XML document was not a representation of that object.

Meltzer does not teach a compilation process of the virtual machine converting the first object into a data representation language representation of the first object. In contrast, Meltzer teaches that after the transaction process has created output, "[t]he output is translated to the format of an output document as defined by the business interface definition." (Meltzer, column 15, lines 12-14). In other words, the output XML document is not a *representation* of any JAVA object, but instead is a *representation of the output data* as defined in an associated business interface definition (BID). Thus, Meltzer is not teaching the creation of a data representation language representation of a JAVA object, but instead is teaching the creation of XML documents, as defined by associated BIDs, that contain data from a JAVA Object.

Additionally, Meltzer also fails to teach a compilation process of the virtual machine converting the first object into a data representation language representation of the first object. Instead, Meltzer teaches that a parser 301 and a translator 302 convert the contents of input XML documents into JAVA objects, and the contents of JAVA objects into output XML documents according to definitions in BIDs. (Meltzer, column 12, line 64 – column 13, line 10). Meltzer also teaches that the output (JAVA objects) from the parser and translator is supplied to the host transaction processing front end and that the output from the host transaction processing front end (also JAVA objects) is supplied back to the translator to be converted into output XML documents. (Meltzer, column 14, line 62 – column 15, line 17). Further, Meltzer teaches that that "[s]uch front ends [host transaction processing front ends] are implemented by JAVA virtual machines." (Meltzer, column 3, lines 62-64, see also column 5, lines 22-23, column 11, lines 31-33, column 13, lines 12-16,).

In other words, Meltzer teaches that only host transaction processing front end is executing in a JAVA virtual machine. The parser 301 and a translator 302 of Meltzer are

not processes of the JAVA virtual machine. Thus Meltzer fails to teach a compilation process of the virtual machine converting an object into a data representation language representation of the object. Hence, Meltzer fails to teach that an object is converted into a data representation language representation of the object, and also fails to teach that a process of a virtual machine performs such conversion.

Applicants assert that the section 102 rejection of claim 1 is not supported by the cited prior art because a rejection under section 102 requires that the identical invention must be shown in as complete detail as is contained in the claims and also requires the presence in a single prior art reference disclosure of each and every element of the claimed invention, arranged as in the claim. (M.P.E.P. § 2131). See also *Lindemann Maschinenfabrik GmbH v. American Hoist & Derrick Co.*, 221 USPQ 481, 485 (Fed. Cir. 1984).

Thus, in light of the above remarks, applicants assert that the rejection of claim 1 is not supported by the cited art and withdrawal of the rejection is respectfully requested. Similar remarks as discussed above in regard to claim 1 apply to claims 40, and 71.

Regarding claim 25, applicants disagree with the Examiner's interpretation of Meltzer and submit that Meltzer fails to teach a method comprising a first virtual machine receiving from a first process a computer programming language object, wherein the object is an instance of a class in the computer programming language; the first virtual machine generating a representation of the object in a data representation language subsequent to said receiving; generating a message in the data representation language, wherein the message includes the data representation language representation of the object; sending the message to a second process; and the second process generating a copy of the computer programming language object from the data representation language representation of the object included in the message.

Specifically, Meltzer fails to teach that a virtual machine generating a representation of the object in a data representation language, as shown above in the

arguments regarding claim 1. Further, Meltzer also fails to teach a second process generating a copy of the computer programming language object from the data representation language representation of the object.

The examiner erroneously characterizes the method recited by applicants' claim 25 as "the object is transmitted over the network and then converted back to XML." This is incorrect. In fact, applicants claim 25 recites, in pertinent part, "generating a message in the data representation language, wherein the message includes the data representation language representation of the object; sending the message to a second process; and the second process generating a copy of the computer programming language *object* from the data representation language representation of the object included in the message." Thus, rather than sending the object and converting the object back into XML, as the Examiner states, a data representation language representation of the object is sent in a message and converted into a copy of the original object. The references cited (Meltzer column 9, lines 24-42, and lines 59-61) by the Examiner refer to devices and processes communicating over various types of networks, including the Internet, but fail to mention anything about converting a data representation language representation of an object into a copy of the object. In fact, applicants cannot find any reference anywhere in Meltzer that teaches the generation of a copy of an object from a data representation language representation of the object.

Thus, in light of the above remarks, applicants assert that the rejection of claim 25 is not supported by the cited art and withdrawal of the rejection is respectfully requested. Similar remarks as discussed above in regard to claim 25 apply to claims 62, and 84.

Regarding claim 34, the Examiner has rejected claim 34 for the same corresponding reasons put forth in the rejection of claim 25 stating, "[i]t has the same inter process exchange of information." Applicants submit, however, that claims 25 and 34 are very different in scope. The Examiner's rejection of claim 34 is improper since it is based on the same arguments as claim 25 even though claims 25 and 34 contain different limitations.

Applicants further note that Meltzer clearly does not anticipate Applicants' claim 34. Specifically, Meltzer fails to teach a virtual machine generating an object from information representing the object. As described above regarding claim 1, Meltzer fails to disclose the generating, in a virtual machine, of an object from a representation of that object. Firstly, Meltzer teaches that JAVA objects are generated from XML documents and then supplied to processes executing in a virtual machine (Meltzer, column 3, lines 62-64, column 5, lines 22-23, column 11, lines 31-33, column 13, lines 12-16). Additionally, Meltzer also fails to teach that the XML documents used to generate his JAVA objects are representation of an object. Just because Meltzer teaches creating an object from an XML document does not mean that the XML document was a representation of the object. In contrast, Meltzer teaches that the XML documents are defined by business interface definitions (BIDs) that comprise "descriptions of sets of storage units and logical structures for the set of storage units" and that are used to validate the proper structure and syntax of the XML documents (Meltzer, column 6, lines 49-52, column 2, lines 55-61). Further, Meltzer clearly states that "output is translated to the format of an output document as defined by the business interface definition." (Meltzer, column 15, lines 12-14). Thus, when Meltzer generates XML output documents from JAVA objects, the resulting XML is not a representation of the JAVA object, but merely contains the information from the object structured as defined in an associated BID.

In other words, rather than being representation of programming objects, Meltzer's XML documents are representation of business-related data as defined by BIDs. Generating JAVA objects to hold the data represented by the XML documents does not constitute generating an object from a data representation language representation of the object.

Thus, in light of the above remarks, Applicants assert that the rejection of claim 34 is not supported by the cited art and withdrawal of the rejection is respectfully requested.

Regarding claim 50, the Examiner has rejected claim 50 for the same corresponding reasons put forth in the rejection of claim 1 stating, "[t]he virtual machine processing instructions is found at column 5, lines 18-31." Applicants submit, however, that claims 1 and 50 are very different in scope and applicants further assert that the Examiner's rejection of claim 50 is improper since it relies upon the same arguments as claim 1.

Applicants also note that Meltzer does not anticipate Applicants' claim 50. Specifically, Meltzer fails to teach a virtual machine receiving a data representation language representation of a first computer programming language object and also fails to teach generating an object from the data representation language representation of the object.

As described above regarding claim 1, Meltzer fails to disclose the generating, in a virtual machine, of an object from a representation of that object. Firstly, Meltzer teaches that JAVA objects are generated from XML documents and then supplied to processes executing in a virtual machine (Meltzer, column 3, lines 62-64, column 5, lines 22-23, column 11, lines 31-33, column 13, lines 12-16). Additionally, Meltzer also fails to teach that the XML documents used to generate his JAVA objects are representation of an object. In contrast, Meltzer teaches that the XML documents are defined by business interface definitions (BIDs) that comprise "descriptions of sets of storage units and logical structures for the set of storage units" and that are used to validate the proper structure and syntax of the XML documents (Meltzer, column 6, lines 49-52, column 2, lines 55-61). In other words, rather than being representation of programming objects, Meltzer's XML documents are representation of business-related data as defined by BIDs. Applicants also argue that generating JAVA objects to hold the data represented by the XML documents does not constitute generating an object from a data representation language representation of the object.

In light of the above remarks, applicants assert that the rejection of claim 50 is not supported by the cited art and withdrawal of the rejection is respectfully requested.

Section 103(a) Rejection:

The Office Action rejected claims 12, 20, 22-24, 78 and 81-83 as being unpatentable over Meltzer in view of Ansari, et al. (U.S. Patent 5,881,290) (hereinafter "Ansari"),

Regarding claim 12, applicants respectfully disagree with the Examiner and submit that Meltzer in view of Ansari fails to teach a virtual machine receiving a data representation language representation of a first computer programming language object from a first process and further fails to teach a decompilation process of the virtual machine generating the first object from the data representation language representation of the first object.

As described above regarding the section 102 rejection of claim 1, Meltzer fails to disclose the generating, in a virtual machine, of an object from a representation of that object. Firstly, Meltzer teaches that JAVA objects are generated from XML documents and then supplied to processes executing in a virtual machine (Meltzer, column 3, lines 62-64, column 5, lines 22-23, column 11, lines 31-33, column 13, lines 12-16). Additionally, Meltzer also fails to teach that the XML documents used to generate his JAVA objects are representation of an object. In contrast, Meltzer teaches that the XML documents are defined by business interface definitions (BIDs) that comprise "descriptions of sets of storage units and logical structures for the set of storage units" and that are used to validate the proper structure and syntax of the XML documents (Meltzer, column 6, lines 49-52, column 2, lines 55-61). In other words, rather than being representation of programming objects, Meltzer's XML documents are representation of business-related data as defined by BIDs. Applicants also argue that generating JAVA objects to hold the data represented by the XML documents does not

constitute generating an object from a data representation language representation of the object.

Ansari teaches a compiler and a decompiler for industrial controllers that allow new instructions to be added to those already recognized by modifying an internal instruction table of instructions. (Ansari, column 2, lines 10-17). Applicants fail to see the relevance of Ansari's method for adding new instructions to an industrial controller compiler to Meltzer's document based business transaction architecture.

The Examiner's cited references in Ansari (column 3, lines 44-47, and column 9, lines 34-38) refer to decompiling an industrial controller program into editable source. However, applicant can find no reference in Ansari referring to generating an object from a data representation language representation of the object, as the Examiner contends. The Examiner seems to be relying upon Ansari merely because Ansari teaches a "decompiler." The Examiner even states this directly, "Meltzer ... does not disclose the translation (of XML into JAVA object) as a *decompilation* process ... [h]owever, Ansari explicitly discloses a method of compiling and decompiling." (emphasis added).

Applicants assert that Ansari's decompiler, which generates industrial controller source code from a program, does not teach generating an object from a data representation language representation of the object. The Examiner even supports this assertion by stating, "the combination [of Meltzer and Ansari] provides a means for the Meltzer invention to extract the object information so that it can be processed as taught [by] Ansari." Applicants submit that the Examiner has failed to show how the combination of Meltzer and Ansari can provide for a virtual machine receiving a data representation language representation of a an object or that teaches or suggests a process of the virtual machine generating the object from the data representation language representation of the object.

Applicants can find no reference in Meltzer or Ansari, separately or in combination, that teaches or suggests a virtual machine receiving a data representation

language representation of a an object or that teaches or suggests a process of the virtual machine generating the object from the data representation language representation of the object.

Thus, in light of the above remarks, applicants assert that the rejection of claim 12 is not supported by the cited art and withdrawal of the rejection is respectfully requested. Similar remarks as discussed above in regard to claim 12 apply to claim 78.

The Office Action rejected claims 2, 26, 29, 52 and 67-68 under 35 U.S.C. § 103(a) as being unpatentable over Meltzer in view of Mitchell (U.S. Patent 6,628,304). These claims are patentable for at least the reasons given above in regard to their respective independent claims.

The Office Action rejected claims 3-7, 42-44, 53-55 and 72-44 under 35 U.S.C. § 103(a) as being unpatentable over Meltzer in view of Humpleman, et al. (U.S. Patent 6,546,419) (hereinafter "Humpleman"). These claims are patentable for at least the reasons given above in regard to their respective independent claims.

The Office Action rejected claims 8, 45 and 57 under 35 U.S.C. § 103(a) as being unpatentable over Meltzer in view of Daly, et al. (U.S. Patent 5,748,896) (hereinafter "Daly"). These claims are patentable for at least the reasons given above in regard to their respective independent claims.

The Office Action rejected claims 13 and 14 under 35 U.S.C. § 103(a) as being unpatentable over Meltzer in view of Ansari and further in view of Mitchell. These claims are patentable for at least the reasons given above in regard to their respective independent claims.

The Office Action rejected claims 15-19, 57, 79 and 80 under 35 U.S.C. § 103(a) as being unpatentable over Meltzer in view of Ansari and further in view of Humpleman.

These claims are patentable for at least the reasons given above in regard to their respective independent claims.

The Office Action rejected claim 21 under 35 U.S.C. § 103(a) as being unpatentable over Meltzer in view of Ansari and further in view of Daly. These claims are patentable for at least the reasons given above in regard to their respective independent claims.

Applicants also assert that numerous ones of the dependent claims recite further distinctions over the cited art. However, since the independent claims have been shown to be patentably distinct, a further discussion of the dependent claims is not necessary at this time.

Information Disclosure Statement:

Applicants note that three additional information disclosure statements with accompanying Forms PTO-1449 were submitted on July 19, 2001, September 17, 2001 and November 18, 2002, respectively. Applicants request the Examiner to carefully consider the listed references and return copies of the signed and initialed Forms PTO-1449 from all three statements.

CONCLUSION

Applicants submit the application is in condition for allowance, and notice to that effect is respectfully requested.

If any extension of time (under 37 C.F.R. § 1.136) is necessary to prevent the above referenced application from becoming abandoned, Applicants hereby petition for such extension. If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5181-72000/RCK.

Also enclosed herewith are the following items:

☒ Return Receipt Postcard

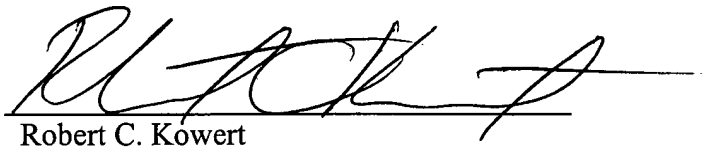
☐ Petition for Extension of Time

☐ Notice of Change of Address

☐ Fee Authorization Form authorizing a deposit account debit in the amount of $ for fees ( ).

☐ Other:

Respectfully submitted,



Robert C. Kowert
Reg. No. 39,255
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8850

Date: June 25, 2004